Date 28/09/21

Neeraj Appari T073

SDS Page No.
Date / /

A I Practical 10

Aim - Write a program to implement Gaussian Mixture Model

Algorithm:
1) Start
2) Import modules (matplotlib, scipy.stats, numpy)
3) use style fivethirtyeight
4) give random seed (0)
5) define X0, X1, X2 to create creature data
6) Combine the Cluster data
7) create array r with dimensionality nxk
8) Instantiate the random gausians
9) Instantiate the random pi_c
10) Probability for each datapoint x_i to belong to gaussian g
11) Normalize the probabilities such that each row of r sums to 1 and weight it by pi_c == fraction of points belonging to cluster c
12) To plot define columns in red, blue and green
13) plot the graph
14) Stop

AI 1o try.py - E:/fffiiles/college pracs and projects/AI/AI 1o try.py (3.8.3)

File Edit Format Run Options Window Help

```python
import matplotlib.pyplot as plt
from matplotlib import style
style.use('fivethirtyeight')
import numpy as np
from scipy.stats import norm
np.random.seed(0)
#Neeraj Appari

X = np.linspace(-5,5,num=20)
X0 = X*np.random.rand(len(X))+10 # Create data cluster 1
X1 = X*np.random.rand(len(X))-10 # Create data cluster 2
X2 = X*np.random.rand(len(X)) # Create data cluster 3
X_tot = np.stack((X0,X1,X2)).flatten() # Combine the cluste



"""Create the array r with dimensionality nxK"""
r = np.zeros((len(X_tot),3))
print('Dimensionality','=',np.shape(r))

"""Instantiate the random gaussians"""

gauss_1 = norm(loc=-5,scale=5)
gauss_2 = norm(loc=8,scale=3)
gauss_3 = norm(loc=1.5,scale=1)

"""Instantiate the random pi_c"""
pi = np.array([1/3,1/3,1/3]) # We expect to have three clus


"""
Probability for each datapoint x_i to belong to gaussian g
```

Python 3.8.3 Shell

File Edit Shell Debug Options Window Help

```
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:20:19)
[MSC v.1925 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for mor
e information.
>>>
======== RESTART: E:/fffiiles/college pracs and projects/A
I/AI 1o try.py =======
Dimensionality = (60, 3)
[[2.97644006e-02 9.70235407e-01 1.91912550e-07]
 [3.85713024e-02 9.61426220e-01 2.47747304e-06]
 [2.44002651e-02 9.75599713e-01 2.16252823e-08]
 [1.86909096e-02 9.81309090e-01 8.07574590e-10]
 [1.37640773e-02 9.86235923e-01 9.93606589e-12]
 [1.58674083e-02 9.84132592e-01 8.42447356e-11]
 [1.14191259e-02 9.88580874e-01 4.48947365e-13]
 [1.34349421e-02 9.86565058e-01 6.78305927e-12]
 [1.11995848e-02 9.88800415e-01 3.18533028e-13]
 [8.57645259e-03 9.91423547e-01 1.74498648e-15]
 [7.64696969e-03 9.92353030e-01 1.33051021e-16]
 [7.10275112e-03 9.92897249e-01 2.22285146e-17]
 [6.36154765e-03 9.93638452e-01 1.22221112e-18]
 [4.82376290e-03 9.95176237e-01 1.55549544e-22]
 [7.75866904e-03 9.92241331e-01 1.86665135e-16]
 [7.52759691e-03 9.92472403e-01 9.17205413e-17]
 [8.04550643e-03 9.91954494e-01 4.28205323e-16]
 [3.51864573e-03 9.96481354e-01 9.60903037e-30]
 [3.42631418e-03 9.96573686e-01 1.06921949e-30]
 [3.14390460e-03 9.96856095e-01 3.91217273e-35]
 [1.00000000e+00 2.67245688e-12 1.56443629e-57]
 [1.00000000e+00 4.26082753e-11 9.73970426e-49]
 [9.99999999e-01 1.40098281e-09 3.68939866e-38]
 [1.00000000e+00 2.65579518e-10 4.05324196e-43]
```

Type here to search

ENG 20:50 28-09-2021

AI 1o try.py - E:/fffiiles/college pracs and projects/AI/AI 1o try.py (3.8.3)

File Edit Format Run Options Window Help

```
"""
Probability for each datapoint x_i to belong to gaussian g
"""
for c,g,p in zip(range(3),[gauss_1,gauss_2,gauss_3],pi):
    r[:,c] = p*g.pdf(X_tot) # Write the probability that x
                            # Therewith we get a 60x3 array f
"""
Normalize the probabilities such that each row of r sums to
cluster c
"""
for i in range(len(r)):
    r[i] = r[i]/(np.sum(pi)*np.sum(r,axis=1)[i])

"""In the last calculation we normalized the probabilites r
to belong to one gaussian (one column per gaussian). Since
to gaussian g, we have to do smth. like a simple calculatio
x_i belongs to gaussian g. To realize this we must dive the
summing up each row in r and divide each value r_ic by sum(
look at the above plot and pick an arbitrary datapoint. Pic
belongs to this gaussian. This value will normally be small
the percentage that this point belongs to the chosen gaussi
gaussian divided by the sum of the probabilites for this da
point belong to each cluster c and threwith to each gaussia
assume that the points are equally distributed over the thr

print(r)
print(np.sum(r,axis=1)) # As we can see, as result each row

fig = plt.figure(figsize=(10,10))
ax0 = fig.add_subplot(111)
```

Python 3.8.3 Shell

File Edit Shell Debug Options Window Help

```
 [9.99999878e-01 1.21709730e-07 1.17161878e-25]
 [9.99999735e-01 2.65048706e-07 1.28402556e-23]
 [9.99999955e-01 4.53370639e-08 2.60841891e-28]
 [9.99999067e-01 9.33220139e-07 2.02379180e-20]
 [9.99998448e-01 1.55216175e-06 3.63693167e-19]
 [9.99997285e-01 2.71542629e-06 8.18923788e-18]
 [9.99955648e-01 4.43516655e-05 1.59283752e-11]
 [9.99987200e-01 1.28004505e-05 3.20565446e-14]
 [9.64689131e-01 9.53405294e-03 2.57768163e-02]
 [9.77001731e-01 7.96383733e-03 1.50344317e-02]
 [9.96373670e-01 2.97775078e-03 6.48579562e-04]
 [3.43634425e-01 2.15201653e-02 6.34845409e-01]
 [9.75390877e-01 8.19866977e-03 1.64104537e-02]
 [9.37822997e-01 1.19363656e-02 5.02406373e-02]
 [4.27396946e-01 2.18816340e-02 5.50721420e-01]
 [3.28570544e-01 2.14190231e-02 6.50010433e-01]
 [3.62198108e-01 2.16303800e-02 6.16171512e-01]
 [2.99837196e-01 2.11991858e-02 6.78963618e-01]
 [2.21768797e-01 2.04809383e-02 7.57750265e-01]
 [1.76497129e-01 2.01127714e-02 8.03390100e-01]
 [8.23252013e-02 2.50758227e-02 8.92598976e-01]
 [2.11943183e-01 2.03894641e-02 7.67667353e-01]
 [1.50351209e-01 2.00499057e-02 8.29598885e-01]
 [1.54779991e-01 2.00449518e-02 8.25175057e-01]
 [7.92109803e-02 5.93118654e-02 8.61477154e-01]
 [9.71905134e-02 2.18698473e-02 8.80939639e-01]
 [7.60625670e-02 4.95831879e-02 8.74354245e-01]
 [8.53513721e-02 2.40396004e-02 8.90609028e-01]]
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
1. 1. 1. 1. 1.
 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
1. 1. 1. 1. 1.
```

Type here to search

ENG 20:51 28-09-2021

AI 1o try.py - E:/fffiiles/college pracs and projects/AI/AI 1o try.py (3.8.3)

File Edit Format Run Options Window Help

```
"""
Probability for each datapoint x_i to belong to gaussian g
"""
for c,g,p in zip(range(3),[gauss_1,gauss_2,gauss_3],pi):
    r[:,c] = p*g.pdf(X_tot) # Write the probability that x belongs to gaussian c in column c.
                            # Therewith we get a 60x3 array filled with the probability that each x_i belongs to one of
"""
Normalize the probabilities such that each row of r sums to 1 and weight it by pi_c == the fraction of points belongin
cluster c
"""
for i in range(len(r)):
    r[i] = r[i]/(np.sum(pi)*np.sum(r,axis=1)[i])

"""In the last calculation we normalized the probabilites r_ic. So each row i in r gives us the probability for x_i
to belong to one gaussian (one column per gaussian). Since we want to know the probability that x_i belongs
to gaussian g, we have to do smth. like a simple calculation of percentage where we want to know how likely it is in %
x_i belongs to gaussian g. To realize this we must dive the probability of each r_ic by the total probability r_i (thi
summing up each row in r and divide each value r_ic by sum(np.sum(r,axis=1)[r_i] )). To get this,
look at the above plot and pick an arbitrary datapoint. Pick one gaussian and imagine the probability that this datapo
belongs to this gaussian. This value will normally be small since the point is relatively far away right? So what is
the percentage that this point belongs to the chosen gaussian? --> Correct, the probability that this datapoint belong
gaussian divided by the sum of the probabilites for this datapoint and all three gaussians. Since we don't know how ma
point belong to each cluster c and threwith to each gaussian c, we have to make assumptions and in this case simply sa
assume that the points are equally distributed over the three clusters."""

print(r)
print(np.sum(r,axis=1)) # As we can see, as result each row sums up to one, just as we want it.

fig = plt.figure(figsize=(10,10))
ax0 = fig.add_subplot(111)
```

Ln: 61 Col: 0

Type here to search

ENG 20:51 28-09-2021

AI 1o try.py - E:/fffiiles/college pracs and projects/AI/AI 1o try.py (3.8.3)

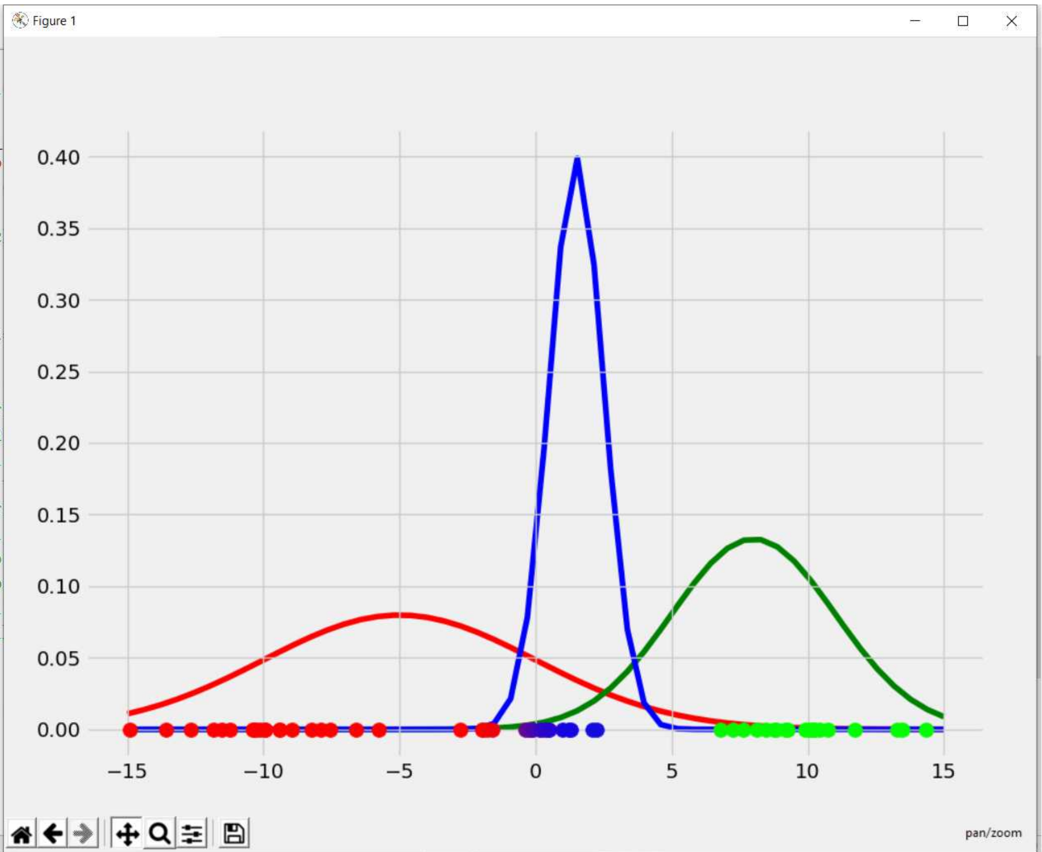
File Edit Format Run Options Window Help

```
"""
Probability for each datapoint x_i to bel
"""
for c,g,p in zip(range(3),[gauss_1,gauss_
    r[:,c] = p*g.pdf(X_tot) # Write the p
                            # Therewith we
"""
Normalize the probabilities such that eac
cluster c
"""
for i in range(len(r)):
    r[i] = r[i]/(np.sum(pi)*np.sum(r,axis

"""In the last calculation we normalized
to belong to one gaussian (one column per
to gaussian g, we have to do smth. like a
x_i belongs to gaussian g. To realize thi
summing up each row in r and divide each
look at the above plot and pick an arbitr
belongs to this gaussian. This value will
the percentage that this point belongs to
gaussian divided by the sum of the probab
point belong to each cluster c and threwi
assume that the points are equally distri

print(r)
print(np.sum(r,axis=1)) # As we can see,

fig = plt.figure(figsize=(10,10))
ax0 = fig.add_subplot(111)
```



Type here to search

ENG 20:51 28-09-2021